REMARKS

The non-final Office Action, mailed January 25, 2006, considered and rejected claims 1-25. Claims 1-6, 8-13, 15-23, and 26 were rejected under 35 U.S.C. 103(a) as being unpatentable over CERT CC, "Cert Advisory CA-2000-02 Malicious HTML Tags Embedded in Client Web Requests" (CERT I) in view of CERT CC, "Understanding Malicious Content Mitigation for Web Developer" (CERT II). Claims 7, 14, and 24 were rejected under 35 U.S.C. 103(a) as being unpatentable over the combination of CERT-Advisory and CERT in view of Fischman et al. (U.S. Patent Publ. No. 2003/0097588).[1]

The Office Action also objected to Figures 1 and 2 of the present application, and requested corrected drawings designating such figures as "Prior Art." Applicants respectfully traverse. In particular, Figures 1 and 2 illustrate exemplary features and environments corresponding to the present invention. For example, Figure 1 illustrates a block diagram of a network environment in which an electronic message may be used in facilitating a cross-scripting attack of a "Hello" HTML page returned from a server to a user computer. The displayed user computer and server can also be construed to include the inventive modules and computer-executable instructions described throughout the application (although they are not explicitly referenced in Figure 1). Nevertheless, one of skill in the art would recognize their presence in view of the disclosure of the application. In addition, Figure 2 is specifically provided as an illustration of the "present invention", as recited in the disclosure of the specification. Nowhere is Figure 1 shown to be or relied upon as prior art. Accordingly, it is respectfully requested that the objections to the drawings be withdrawn.

By this paper, claims 1, 7, 8 and 18 have been amended, while no claims have been added or cancelled.[2] Following this paper claims 1-25 remain pending, of which claims 1, 8 and 18 are the only independent claims at issue.

Applicants' claimed invention is directed to systems, methods and computer program products in which a server mitigates cross-scripting attacks which would otherwise present non-server data in the context of server data. As recited in claim 8, for example, an HTTP request is

[1] Although the prior art status of the cited art is not being challenged at this time, Applicants reserve the right to challenge the prior art status of the cited art at any appropriate time, should it arise. Accordingly, any arguments and amendments made herein should not be construed as acquiescing to any prior art status of the cited art.

[2] Support for the claim amendments can be found throughout the specification, including, but not limited to, the disclosure found in originally filed paragraphs [0007] and [0025].

received by a server computer and includes input data that was not generated by the server computer. The HTTP request is then evaluated to determine if the input data includes a script construct indicating that the HTTP request is part of a cross-site scripting attack. If a script construct is found, the server refuses to execute the HTTP request, thereby preventing the cross-site scripting attack. In addition, a response is generated that indicates a script construct representing a cross-site scripting attack has been received, and resubmission of the HTTP request is requested. Inasmuch as each incoming HTTP request may be evaluated, the resubmitted HTTP request is then executed only upon determining that it does not contain the script construct.

Claims 1 and 18 are directed to a method and computer program product, respectively, and generally correspond to the method of claim 8.

While the cited CERT I and CERT II references generally relate to preventing the insertion of malicious scripts in an HTTP request, they fail to disclose or suggest the present invention. For example, among other things, CERT I and CERT II fail to disclose or suggest generating a response indicating that a script construct has been received and requesting resubmission of the HTTP request, as recited in combination with the other recited elements.

In fact, CERT I and CERT II also fail to disclose executing the HTTP request submitted to the server. In particular, CERT I generally discusses the problem associated with malicious code inserted in cross-site scripting attacks. (p. 1-2). CERT I further notes that users may take precautions to solve this problem (e.g., by disabling scripting or selectively visiting web sites). (p. 4, ¶¶ 4, 5). In addition, CERT I cursorily notes that web site developers can prevent abuse of the site by allowing only a limited character set and by encoding and filtering data. (p. 5, ¶¶ 4-6). For the details on encoding and filtering, however, CERT I defers to the CERT II document.

CERT II provides five steps to mitigate the vulnerability of presenting data as originating from that server, even if it does not. In particular, CERT II describes these steps as: (i) explicitly setting the available character set; (ii) identifying special characters; (iii) encoding dynamic output elements; (iv) filtering specific characters in dynamic elements; and (v) examining cookies. (p. 2, ¶ 3). CERT II specifically identifies a variety of characters (i.e., special characters) which have an effect on how a page is displayed (e.g., greater-than and less-than signs, ampersand, double quotation marks, space, tab, new line, percent, semicolon, non-ASCII characters, curly braces, exclamation point, etc.). (p. 3). By understanding which special

characters affect a web page, at the time data is output (or input), the web page is filtered to detect these special characters. (p. 4, ¶¶ 3, 4). Upon detection of these special characters, the system removes the special characters in the HTTP request. (*See* pp. 5, 6, C++, Javascript, and Perl examples).

Accordingly, CERT I and CERT II disclose that when an HTTP request is received, the character set should be limited and special characters should be removed. In this manner, the HTTP request can be executed without giving any significance to the now-removed special characters. In contrast, however, to CERT I and CERT II which process the HTTP request (albeit without the special characters), Applicants specifically recite refusing to execute the HTTP request so as to prevent the scripting attack and requesting that the request be resubmitted. In other words, CERT I and CERT II prevent the scripting attack by processing the page without the special characters, while the claimed invention refuses to execute the HTTP request until it is resubmitted.

In addition, CERT I and CERT II also fail to disclose or suggest generating a response indicating that a script construct indicative of a cross-site scripting attack has been received and requesting resubmission of the HTTP request which is subsequently executed if it does not contain the script construct, as claimed in combination with the other recited elements. In fact, inasmuch as CERT I and CERT II appear to disclose filtering portions of the request and then executing the HTTP request after it is received, Applicants respectfully submit that it would be irrational to request resubmission of the HTTP request for subsequent execution.

As will be appreciated, requesting that a user resubmitted an HTTP request and notifying the user that the HTTP request includes an unauthorized script can provide significant benefits to the user not provided by the disclosure in the CERT I and CERT II references. For example, where the user, after selecting the link, is notified that the link is an attack, the user will alerted that the link was provided by an attacker rather than by an authorized source. As a result of this knowledge, the user can better detect links with such script constructs in the future, and avoid emails or web pages from the attacker, avoid executing links provided by the attacker, block the specific attacker, or act in a variety of other ways to protect against attacks. While removing special characters may work to prevent the attack, it does not necessarily provide the potential benefit of educating the user as to the nature and existence of the attack.

In view of the foregoing, Applicant respectfully submits that the other rejections to the claims are now moot and do not, therefore, need to be addressed individually at this time. It will be appreciated, however, that this should not be construed as Applicant acquiescing to any of the purported teachings or assertions made in the last action regarding the cited art or the pending application, including any official notice. Instead, Applicant reserves the right to challenge any of the purported teachings or assertions made in the last action at any appropriate time in the future, should it arise. Furthermore, to the extent that the Examiner has relied on any Official Notice, explicitly or implicitly, Applicant specifically requests that the Examiner provide references supporting the teachings officially noticed, as well as the required motivation or suggestion to combine references with the other art of record.

For at least the foregoing reasons, Applicants respectfully submit that the pending claims are neither anticipated by nor made obvious by the art of record. In the event that the Examiner finds and remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney.

Dated this 25 day of April, 2006.

Respectfully submitted,

RICK D. NYDEGGER
Registration No. 28,651
JENS C. JENKINS
Registration No. 44,803
Attorneys for Applicant
Customer No. 047973

RDN:JCJ:CCN:ppa
PPA0000003226V001